Acquiring and Processing Chemical Maps

Matthew Marcus

April, 2008

0. Introduction

Chemical mapping is the process of mapping not only the concentration but the chemical state (valence, usually) of an element. It relies on the fact that the XANES of an element varies with chemical state. Therefore, if one does XRF at several energies near the absorption edge, it is in principle possible to extract the concentrations of various species provide that the sample contains only known species or species having very similar XANES spectra. Figure 1a shows the XANES spectra for a number of Fe species. These are the species used in the map which will be used as the worked example in this manual. The sample is a Stardust track, and the species are "generic" metal, sulfide, Fe^{2+} oxides (silicates) and Fe^{3+} oxides, each of which is actually a weighted sum of the species found by doing point XANES analyses on many particles. For the purposes of this example, we can pretend that these are single compounds.

We see that each species has distinctive XANES features. By probing at energies such as shown by the vertical dotted lines (7110, 7113, 7117.5 and 7126eV), one can make maps in which different species are excited with different sensitivities. Mapping at an energy far above the edge (7210.75eV = [edge]+100eV) yields the total amount of the element, in this case Fe. Mapping just below (7100eV) yields the background signal. If there is a part of the sample devoid of the element of interest, one can take a background XANES

spectrum and consider that as another species. Figure 1b shows the pre- and post-edge energies and the background "species" spectrum.

Thus, a chemical map dataset consists of several maps, each of the same area, but taken at different energies. The task, then, is to acquire a useful set of such spectra and to analyze them.

The theory goes something like this [Sutton, Bajt, Delaney, Schulze, Tokunaga, *Rev. Sci. Inst.* **66**, 1464 (1995)]: Let the normalized XANES signals from each species be $Y_i(E)$ for the ith species, with all Y_i going from 0 far below to 1 far above the edge. Then, if the sample contains amounts x_i of each species, the signal as a function of energy is

$$Y(E) = \sum_{i=1}^{N} Y_i(E) x_i \qquad (1.1)$$

with N the number of species. If the maps are taken at energies $E_a, a = 1 \ldots M$ with $M \geq N$ the number of maps, we get

$$I_a \equiv Y(E_a) = \sum_{i=1}^{N} Y_i(E_a) x_i = \sum_{i=1}^{N} Y_{ai} x_i \qquad (1.2)$$

or in matrix notation

$$\vec{I} = \overline{\overline{Y}} \vec{x} . \qquad (1.3)$$

Note that this equation is overdetermined if $M \geq N$ so it must be solved in a least-squares sense with a non-negativity constraint $x_i \geq 0 \, \forall i$. In our program, a standard SVD algorithm is used to solve the least-squares problem, and if any resulting x is negative, the best fit with some x's constrained to be 0 is taken. Shot noise is taken into account by assigning a standard deviation to each pixel depending on the counts in that pixel. Any

background spectrum is normalized so that it comes to about 1 in the post-edge region and is treated as just another species. If there is no such spectrum, than an artificial one is used for which $Y_b = 1$ for all energies.

In this manual, it is assumed that the reader is familiar with the programs for XRF processing including registration, deadtime correction, data display and making composite maps. These are all documented in other manuals.

I.   Acquisition

The first step is to choose a set of energies at which to take the data. If there are only two species, e.g. Cr^{3+} and Cr^{6+}, then a good way to go is to use Plot-Add-Multiply to plot the ratio of the normalized XANES spectra and pick one of the energies to be the maximum or minimum. In the case of Cr, the ratio of the signal from Cr^{6+} to that of Cr^{3+} will have a maximum near the peak of the prominent chromate pre-edge feature. That pins one energy. As a minimum, one would use another energy just below the pre-edge features of either species, and another 100-200eV above the edge, at which the EXAFS wiggles have somewhat died out. In general, one needs at least one more energy than species, in order to be able to account for the pre-edge background. In the Fe example, two of the energies (7113eV and 7117.5eV) were chosen because it has been found that the intensities at this pair of energies, when plotted against each other, give good discrimination between classes of compounds [Marcus, Westphal, Fakra, sub. *J. Synchrotron Radiation*].

In Figure 1, it is apparent that much of the data is taken at energies at which the signal from the Fe is small compared to what it is at the post-edge. Therefore, it may be useful to take the lower-energy maps with a longer dwell time than the high-energy maps.

In the worked example, the dwell times range from 500-200ms/pixel. A new program has been developed to help make rational choices of energies and dwell times and is described below. This program is as yet too new for its true value to be known.

It's also useful for the pixel size to be smaller than the beam size, if that's practical, especially if the sample contains small particles. The reason is that the maps taken at various energies usually have to be registered against each other because the sample moves slightly, and this process causes intensity to spill from each pixel to adjacent ones. This spillage is an inevitable result of interpolating in fractional pixels. If the intensity from a particle only shows in one pixel and the corresponding map must be shifted by half a pixel to register it with another map, then the resulting registered map will exhibit signal at two adjacent pixels. If this is then compared with the map taken at another energy for which no such smearing took place, the linear algebra involved in extracting species amounts won't work correctly. In a tricolor map, this artifact appears as color fringing. Oversampling the data by taking it with a pixel size of 1/2-2/3 of the beam size reduces this problem. In the example, the spot size is 5μm and the pixel size 3μm.

II.   Initial reduction

Of course, the data should be deadtime corrected. Normalization is particularly important for chemical maps because the results depend on delicate cancellations and small differences of large numbers. I find it convenient to save the files into another directory which is only for deadtime-corrected maps, and under filenames which include the energy, such as `track104 carcass chem 7113 DT.xrf`. Similarly, at each

stage of processing, it's nice to make a new directory to reduce the chance of applying an operation to a file which is at the wrong stage of processing.

Next, it's helpful to resize the images to reduce further the effect of registration artifacts. The program to do this is called Resize XRF Map. Simply invoke the program, give it the file, and, as shown in Figure 2, tell it the factor by which you want to multiply the size of the image. In this case, we multiply both X and Y sizes by 2. Thus, we enter 2 into the bottom control and push the green button next to that control. That sets the new size. Push the big red Enter button and answer the file dialog.

The next step is registration, which is done as one would for composite maps. For chemical mapping, choose one map to be the reference to which the other are registered. In some cases, there will be an element whose edge energy is lower than the incident for any of the maps, and which is present as distinct features. When this condition is met, one can pick any of the maps to be the reference and register using the lower-energy element. In the Stardust case, there was no such element, and the best we could do was to register to the Fe channel at 7117.5eV, a medium energy.

Next, the maps taken at all the different energies need to be brought together into one map. Further, the chemical-map analysis program identifies the energy for each map by reading the channel name and looking for a number of appropriate magnitude. Thus, one wants to make a composite map out of all the registered map, in which the channels have names like 7110 and 7117.5. Here is the header of the composite map for the Stardust example:

```
Title   track104 carcass chem
X scan range (um) -17802.020000       -18474.280000
Y scan range (um) 11989.480000        11710.480000
Scan speed (um/sec) 6.002232
# points, # scan lines 449 185
Scan steps X,Y -1.500000        -1.500000
Dwell time 500.000000
Energy (eV) 7100.009514
# of SCAs 6
SCAs - lower, upper, name
615    669    7100
615    669    7110
615    669    7113
615    669    7117.5
615    669    7126
615    669    7210.75
Date&Time: 2/6/2008      3:06 AM
Start of data block
Data array size: 6        185    449
```

Here, we have used Mix&Match Maps to extract only the Fe channels (originally named

FeKa/MnKb) from each map and combine them. The easiest way to do this is to start

with one of the maps (say, the 7100eV one) as both donor and main map (see the manual

on Registering XRF images if you don't understand any of this), then toss all channels

from the main map and transfer the appropriate channel from the donor, calling it 7100.

Read in a new donor file and repeat the transfer process, naming the channel according to

the energy.

III.   Chemical map

Next, we have to do the linear algebra described in the Introduction. For this, we

need the normalized XANES signals called out in the math as Y_i. These would be *.e-

type normalized XANES files. The chemical-map program reads in a *.prm file which

lists the *.e files in the same was as is done in the linear-fit programs. In our Fe-in-

Stardust example, the file looks like this:

```
ref=Track74b background FeXANES smoothed.e
ref=generic sulfide for chem map.e
ref=generic metal for chem map.e
ref=generic fe2+ for chem map.e
ref=generic fe3+ for chem map.e
Ref = ""
```

and the `*.e` files live in the same directory as the `*.prm` file. As in the linear-fit programs, you can specify a full path name instead of just a base filename. The program which does the operation is Chemical map. When it starts up, it asks if you want to use a constant background. If you have a background spectrum, as in this example, say no. If you say yes, it will generate a constant background as a "species". After reading in the composite file and the `*.prm` file, it grinds for a while, then spits out a new map, suggesting a filename with "chem_map" appended. Here's the header of the map:

```
Title  track104 carcass chem
X scan range (um) -17802.020000     -18474.280000
Y scan range (um) 11989.480000      11710.480000
Scan speed (um/sec) 6.002232
# points, # scan lines 449 185
Scan steps X,Y -1.500000      -1.500000
Dwell time 500.000000
Energy (eV) 7100.009514
# of SCAs 13
SCAs - lower, upper, name
615   669   track74b background fexanes smoothed.e
615   669   generic sulfide for chem map.e
615   669   generic metal for chem map.e
615   669   generic fe2+ for chem map.e
615   669   generic fe3+ for chem map.e
615   669   7100
615   669   7110
615   669   7113
615   669   7117.5
615   669   7126
615   669   7210.75
615   669   MSE
615   669   Norm. MSE
Date&Time: 2/6/2008      3:06 AM
Start of data block
Data array size: 13      185    449
```

The first 5 channels are the derived amounts, and they're shown in Figure 3. The white levels in those figures are 1000 for the first 3 channels and 2000 for the $Fe^{2,3+}$. We see that there's some sulfide, Fe^{2+} and Fe^{3+} but essentially no metal. The brighter area in the background map is where the aerogel block is. Point XANES spectra on some of the spots shown in this map bear out the identification. The next six channels are the original ones from the composite map used as input for the chem-map program. The next ones are the mean-square ($\langle(I - I_{fit})^2\rangle$) and normalized mean-square errors ($\langle(I - I_{fit})^2\rangle / \langle I^2\rangle$). These are scaled by factors of 100 and 10^5, respectively (by default) so as to avoid roundoff truncation due to the integer format of the data block of $\texttt{*.xrf}$ files. If the number of maps equals the number of species (counting the background "species") and the amounts of all species are positive, then equation (1.3) is exactly satisfied and the mean-square error is 0.

IV.   Error analysis

How good is the chemical map likely to be? What energies should we use? Is it better to count more at the low energies than at higher energies? To answer these questions, we need some means of estimating the error of the chemical map. A first approximation to this can be obtained by assuming that the sample is well-described as a sum of the given references and that the noise is entirely Poisson noise. Now, we can then make up a fake data set consisting of counts for the given energies, and with standard deviations equal to the square root of those counts. On feeding this into the least-squares routine, we get a variance estimate based on the covariance matrix. Now, the standard error of the "data" depend on that data, so we should simulate a variety of compositions. We therefore sample a grid of compositions such that the amounts of the

species always add up to 1. We then do the calculation for each of these simulated compositions and average the resulting standard errors for the species. Thus, if we have three species, we might have compositions like (1,0,0),(0,1,0),(0,0,1),(1/2,1/2,0), (0,1/2,1/2),(1/2,0,1/2). The result of error estimation is a set of six standard deviations for each of the three component amounts. We average over the six, leaving error estimates for the three components. This process runs fast enough so that it's easy to make it interactive, so the user can see what the effects are, for instance, of changing the energies.

An examination of the main screen of the error estimator may make it clearer how to use it (Figure 4). On entry, the program asks for the database file, which should be the same `*.prm` file you'll use to analyze the map. As usual, this file will point to a set of reference spectra. A graph similar to Figure 1b will appear showing all the spectra. This program does not ask whether you'd like a constant background, so if you want an artificial background, make a file with the appropriate abscissa range and all 1's in the ordinate.

By default, the program will assume that there will be as many maps as references and that the energies will be equally spaced across the range spanned by the references. This is usually not what you want. You can move the cursors around the graph to get different energies, and you can add cursors. You can also read energies from an existing composite map file, as was done before the screen shot in Figure 4 was taken.

The time/pixel for each map is entered in the bank of green controls to the right of the green 'Energies' indicators. There's another control that's a bit tricky. Since the different maps will tend to have different count rates just because of where they fall on

the XANES curves, I put in a common normalizer which is the number of counts/second in the bright pixels at a post-edge energy. Thus, suppose there is one species at a given point, and this counts number is set at 1000. Suppose further that the energy for one of the maps is such that the XANES signal from this species is 50% of the post-edge value. Then, the count rate expected for that pixel and that map will be 500/sec. You can estimate what the count-rate number should be by finding a typically-bright pixel and measuring the count rate well above the edge, say 100eV above.

With that, you can read off two sorts of error estimates. One is the error expected for the total amount of the given element, which in this case is 3.68%. That means that the measurement you get by adding up all the species should be good to about 3.68%. This error bar does not differentiate between the background and real species.

The other error bar is shown in the fill indicators to the right of the 'Names' indicators. These show the error expected in the fraction of absorbing atoms to be found in the given species. In this example, the error figure for sulfide is 0.13, so you can't be sure if a given pixel is 63% or 37% sulfide. In practice, the chemical maps look better than estimated by this method, though maybe they aren't. These error estimates do not take the non-negativity constraint into account. You can move the energy cursors around and see what happens to the error bars. If you have exactly as many energies as species, and move one cursor until it lands on another, you can see the error bars go crazy because you've made the problem singular.

The Simulated tab shows the simulated XANES at each energy point and its error envelope for one of the compositions. You can scroll through the compositions to see how it changes.

The Test Reference button brings up a new window which allows you to see what happens if what's in the sample isn't exactly what's in the database. Figure 5a shows what happens in our Stardust example when the pixel contains pyrite, which isn't all that well represented by the generic sulfide (mostly pyrrhotite and similar). In this case, it tries to match by adding in negative amounts of some components. This may be what will happen in actual data if the pixel contains enough of these components so that the total adds up to something non-negative. The other extreme is if the fractions are constrained to be non-negative, which is done using the toggle switch on the right. Now, we get a much worse fit, but the pixel still appears to be pure sulfide. Pyrrhotite, which is more similar to "generic sulfide" than pyrite is, yields a much better fit, but at the cost of appearing to contain some Fe^{2+}. Thus, we can use this tool to see what kinds of artifacts might result from using an inadequate reference set.
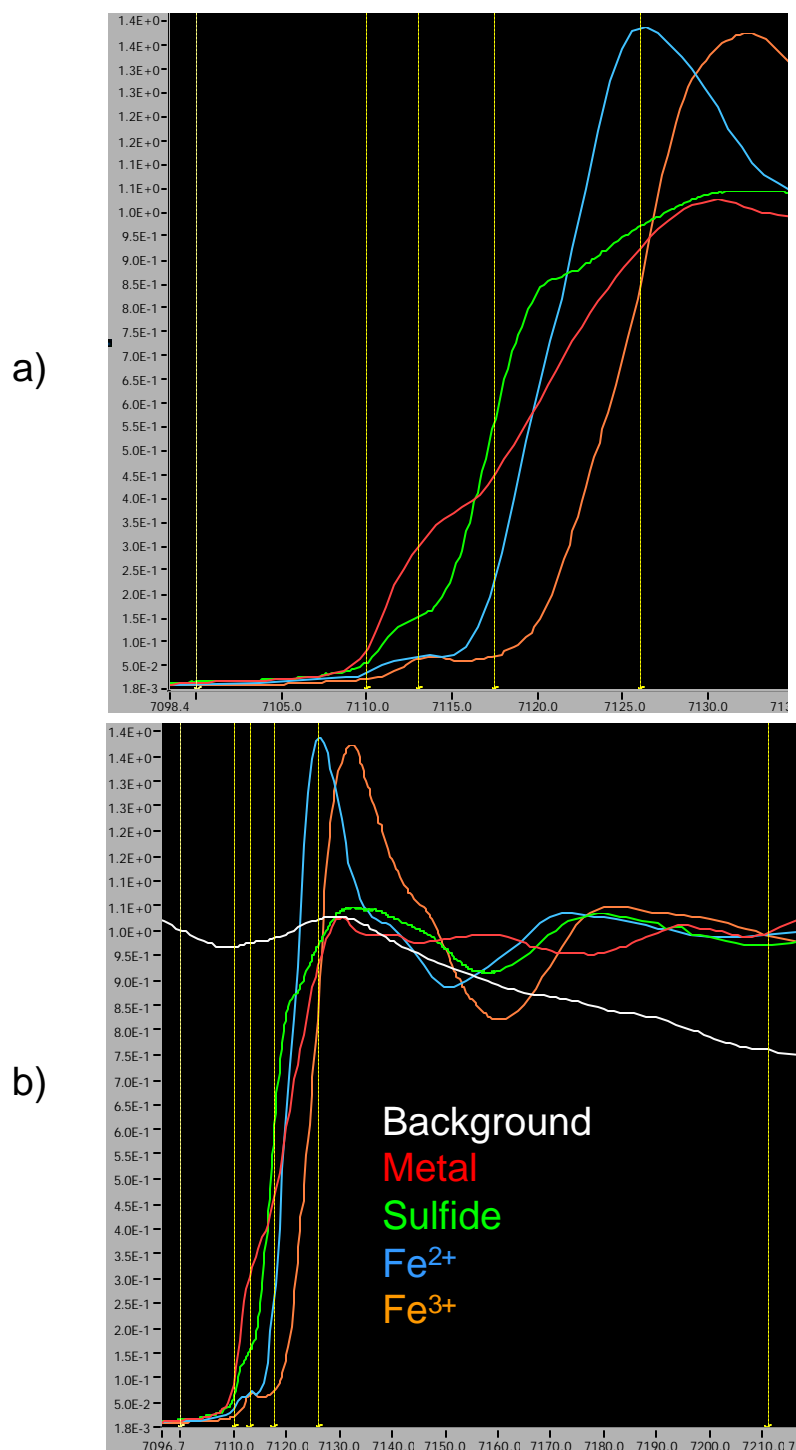
Figure 1. XANES spectra of Fe-bearing references in Stardust example. Dotted vertical lines show the energies at which maps were taken.
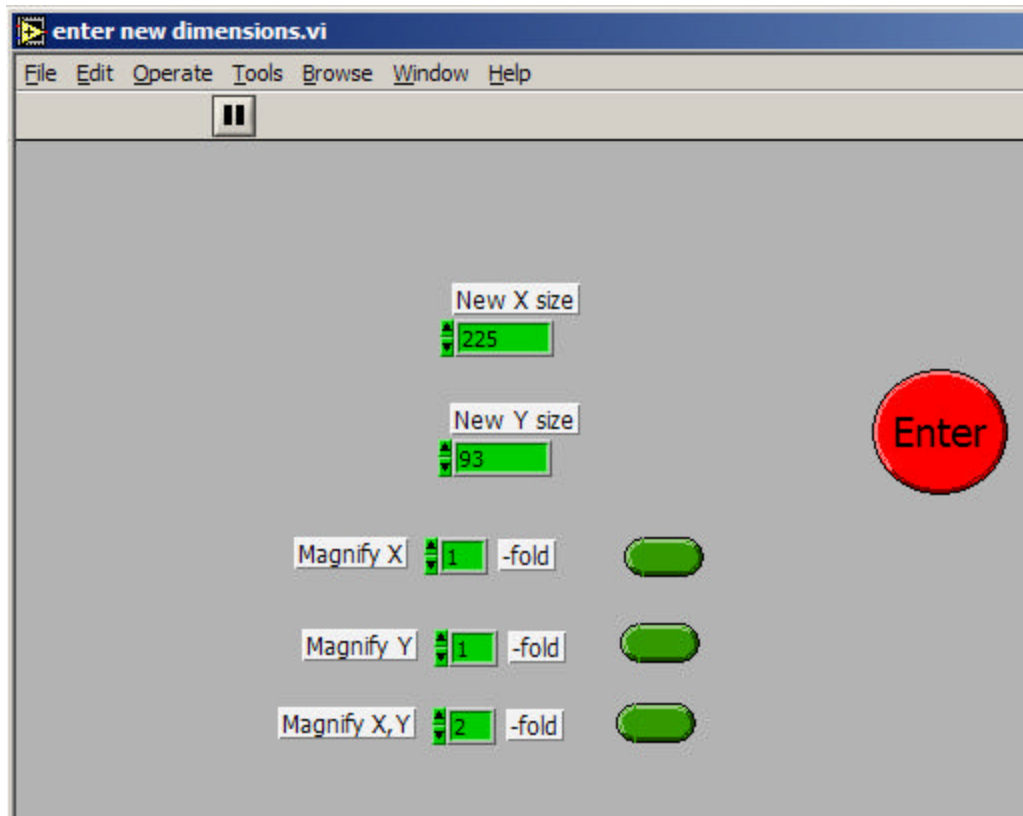
Figure 2. Screenshot of size-entry screen for resizing a map. In this case, pushing the green button opposite the Magnify X,Y label sets the size, and pushing the Enter button makes it happen.
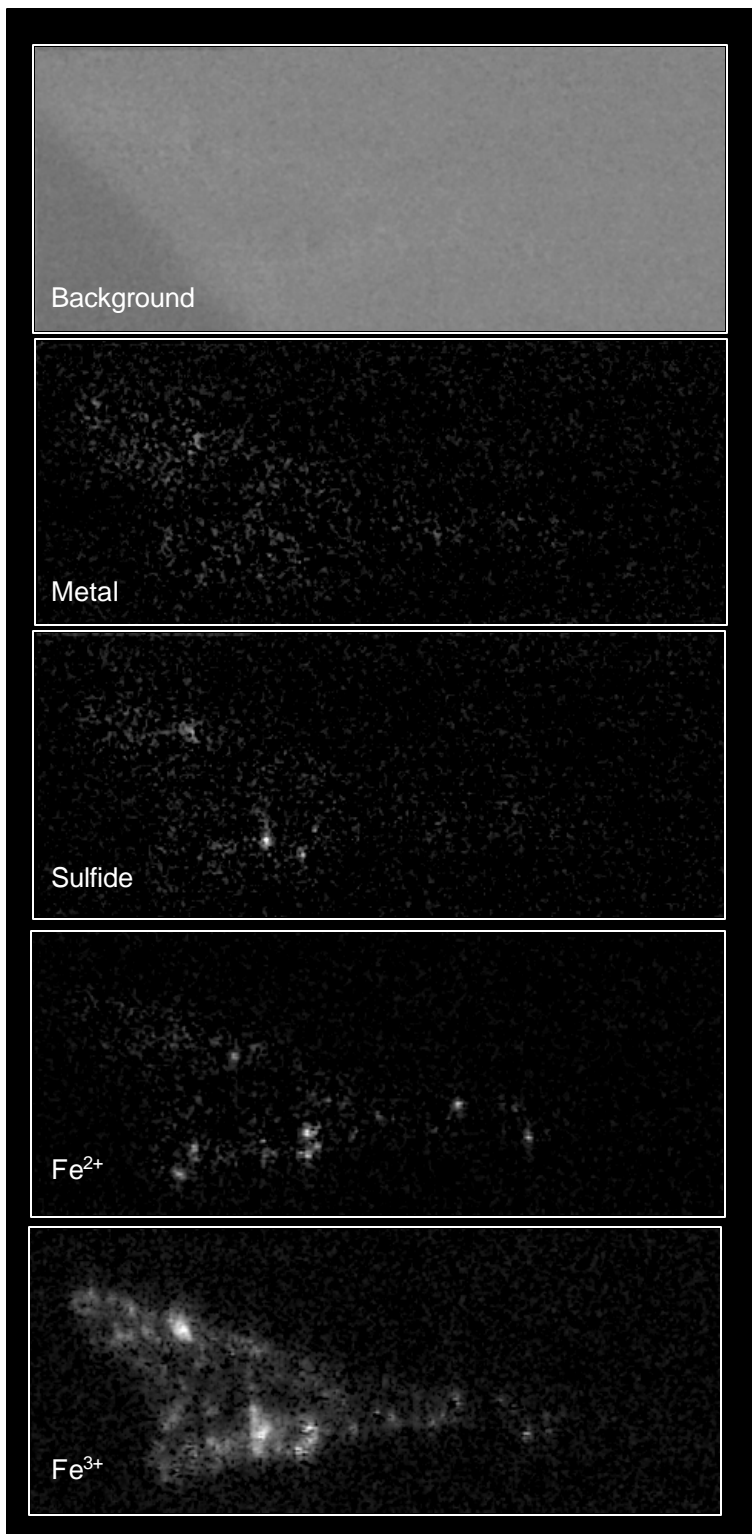
Figure 3: The chemical maps in the Stardust example. The white levels are 1000 for the top three maps and 2000 for the bottom two.
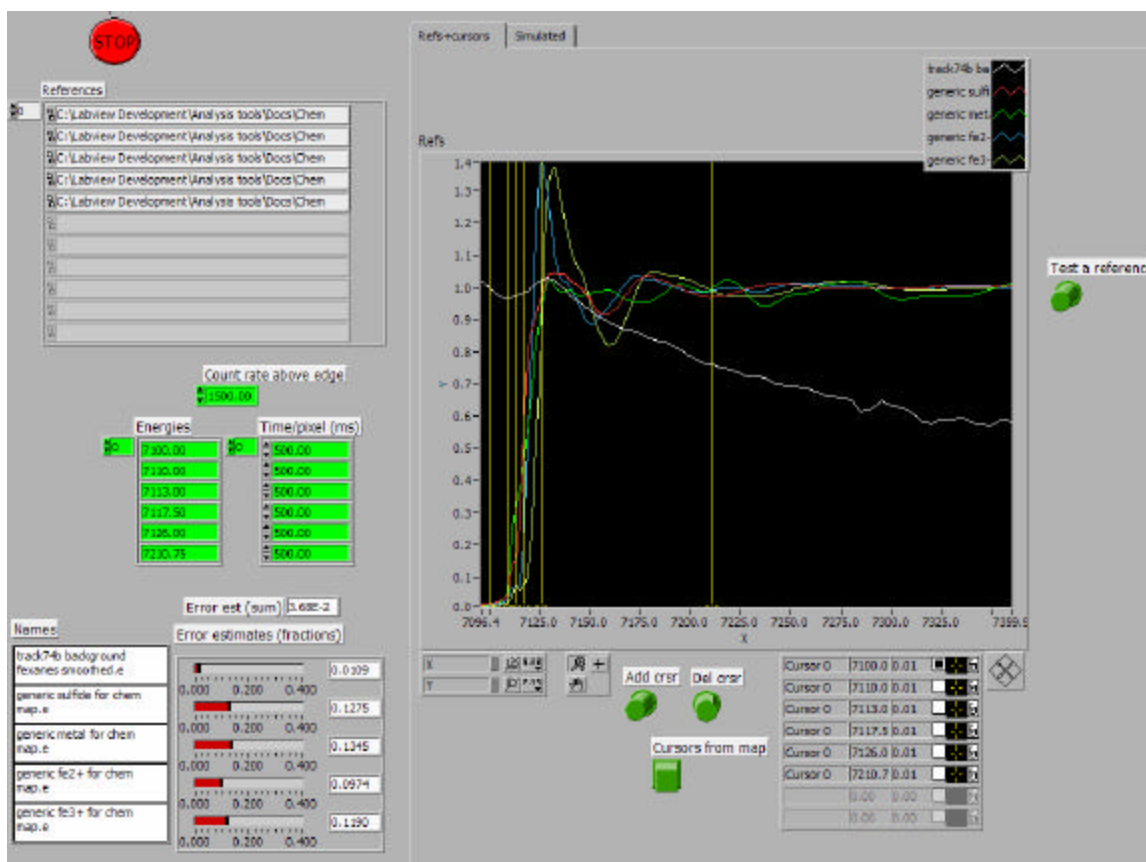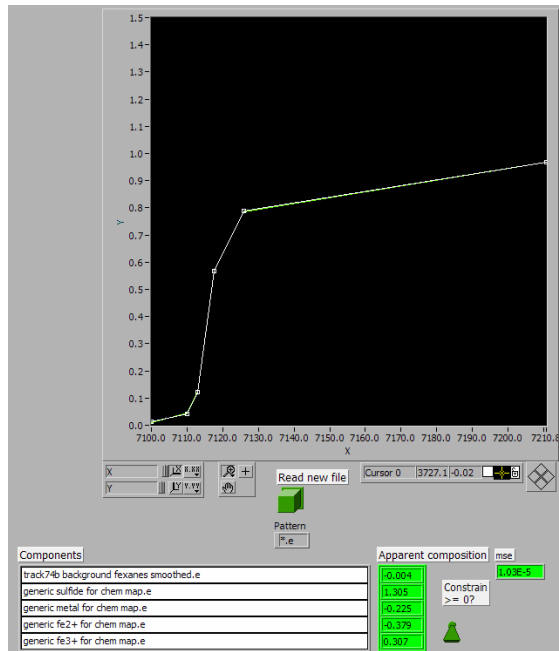
Figure 4: Screenshot of error estimator program showing the references (including background) and an example of error estimation.
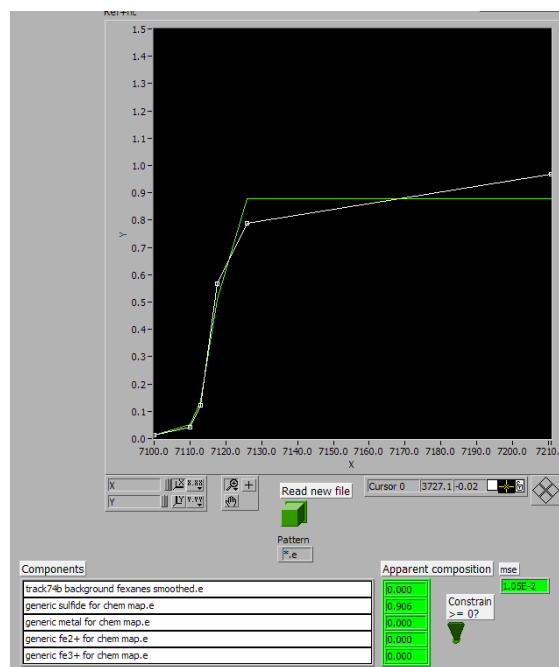
Figure 5: Testing to see how a particle of pyrite would appear in the chemical mapping. In a) the non-negativity constraint is not imposed, and a good fit results from using negative amounts. In b) the constraint is imposed, and the fit is worse, but at least it shows the pyrite to be a sulfide.